**Esercizio 1 - Flywheight pattern - Stampante a colori**

Requisito:

Lo scopo è realizzare il Firmware di una stampante che esegua la stampa di un testo a colori, avvalendosi del pattern Flyweight per ridurre al minimo l’occupazione di memoria di ogni singola sessione di stampa

Il testo da stampare (con i relativi colori) è il seguente

**Hello** **World !**

La simulazione dell’output della stampante sarà affidata alla Console della vostra applicazione.

Suggerimenti:

1. Ogni carattere può essere modellato da una classe che contenga:
   * Il carattere stesso
   * Il colore 🡪 emumerato ConsoleColor
2. Il Flyweight permette di istanziare un solo oggetto per ogni tipo di combinazione carattere-colore, anche qualora lo stesso appaia più volte all’interno del testo

**Esercizio 2**

Requisito:

L’esercitazione punta a creare una sorta di piccolo videogico. Lo scopo dell’applicazione sarà:

* Creare tre personaggi
* Assegnare delle armi ai personaggi
* Ogni personaggio può portare solo un’arma per tipo. Qualora raccogliesse la stessa arma più volte, le munizioni dovranno essere incrementate le munizioni di 10 unità
* Stampare il nome del personaggio e l’arma equipaggiata (ovvero quella del primo slot)

Di seguito la lista dei personaggi e dei tipi di armi

public enum HeroClass

{

Tank,

Damage,

Support

}

public enum WeaponType

{

Knife,

Gun,

AssaultRifle,

SniperRifle

}

**Esercizio 3**

Riallacciandosi all’esercizio visto in precedenza, agire sulle classi concrete che modellano i player assegnando dei comportamenti

In sostanza ciascuna classe personaggio dovrà esporre i metodi che permettono di eseguire i seguenti comportamenti

* Muoversi
* Correre
* Sparare con l’arma equipaggiata

La classe “Support” dovrà implementare un’azione specifica

* Cura un altro personaggio

**Esercizio 4 – API RESTful**

Crare una applcazione MVC utilizzando l’IDE Visual Studio, che permetta di interagire con il gioco utilizzando un set di API RESTFul.

L’applicazione dovrà esegure i seguenti compiti

* Creare un sessione di gioco 🡪 <http://localhost:1234/Game/Create> e restituisca un id partita
* Esporre una API che elenchi la lista dei personaggi 🡪 <http://localhost:1234/Game/Heroes>
* Esporre una API che visualizzi le statistiche di un singolo personaggio 🡪 [http://localhost:1234/Game/Hero/{id}](http://localhost:1234/Game/Hero/%7bid%7d)
* Esporre una Api che esegua faccia esegure un’azione scelta ad uno dei personaggi e ritorni un messaggio di conferma per l’avvenuta esecuzione dell’azione 🡪

[http://localhost:1234/Game/Hero/{id}/{action}](http://localhost:1234/Game/Hero/%7bid%7d/%7baction%7d)